CLUBE

Z80

/83

N.º11

COORDENADORES: MARIA IRENE • ALBERTO FERNANDES • AV. BOAVISTA, 832 • TELEF. 65127 • 4100 PORTO

CLUBE Z-80

NUMERO 11 / 31 AGOSTO 1983
AV. BOAVISTA, 832 - 2 T. Telef.65127
4100 PORTO

N E S T E   N U M E R O

==

I N T R O D U C A O   A   L I N G U A G E M   M A Q U I N A
(Continuacao)

==

FERNANDO D'ALMEIDA PRECES

PROGRAMA 3a - Transferencia duma ROTINA, TABELA ou DADOS da ROM para a RAM, utilizando a mesma instrucao.

Para que um tal trabalho? Poderia perguntar o leitor, pois a primeira vista parece realmente um trabalho inutil. No entanto, se um utilizador da maquina tiver uma rotina da ROM numa zona da memoria aonde lhe possa mexer (modificar o sentido, o salto, etc.), possivelmente encontrara muitas respostas, tirara muitas duvidas e talvez - quem sabe? - descubra algo sobre a maquina, o seu comportamento ou formas de programacao que ate agora nao tenham sido experimentadas.
Por este lado tem, sem duvida alguma, um vasto campo de investigacao a sua espera. Experimente sem receio e lembre-se que alguns avancos mais notorios da tecnica de computacao nao foram criados por grandes sumidades em electronica, mas sim por leigos como nos.
Para executar esta experiencia, basta somente na ROTINA CM 1 carregar o registo HL com a localizacao pretendida e carregar BC com o numero de bytes a transferir.
E claro que muitos utilizadores do ZX nao devem saber aonde se iniciam e terminam as rotinas da ROM, mas nisso vamos nos tentar ajuda-los, fornecendo-lhes um indice de algumas rotinas que ainda nao foram referenciadas neste texto.

Eis algumas como exemplo:

A Rotina de comando SAVE (02F6 a 033E)
sub rotinas (01FC a 01FE e 0201 a 0207)

A Rotina de comando LOAD (0340 a 03A7)

A Rotina de pesquisa do teclado (02BB a 02E6)

A Rotina de descodificacao do teclado (07BD a 07DB).

E para finalizar este capitulo vamos executar um programa (o 3b), que nos permita obter no ecran imagem em movimento.

IMAGEM ANIMADA

Ao longo deste capitulo temos introduzido as varias rotinas em codigo maquina na primeira linha de programa, apos uma instrucao REM. Como ja foi afirmado no inicio, este sistema e um dos mais utilizados neste computador, por ser muito facil a sua aplicacao, mas enferma de algumas desvantagens entre as quais talvez a mais importante seja o facto de provocar o colapso do sistema BASIC que implica, se nao corrigido de imediato, o desmoronamento do programa ao utilizar-se como instrucao C.M. o codigo 118, que dentro da textura BASIC ZX81 representa uma instrucao (salto para uma nova linha)

NEW LINE.

Com a introducao deste caracter, as duas instrucoes imediatas sao consideradas como um numero composto (2 bytes) e enviadas para a rotina ROM - ESCREVA UM NUMERO DECIMAL COMPREEENDIDO ENTRE 1 E 9999 - o que fara surgir uma nova linha BASIC inserida entre a fragmentada e a linha seguinte do programa.

O numero de ordem dessa nova linha depende do valor decimal dos dois bytes, segundo a formula ja nossa conhecida que estabelece ser o numero de linha igual ao valor decimal do primeiro byte multiplicado por 256 e somado ao valor do segundo byte.

Outro fenomeno nao menos catastrofico para o programa vai ocorrer dentro dessa mesma linha. Os dois bytes seguintes devem indicar o numero de bytes contido nessa linha de instrucoes e so por uma grande coincidencia isso podera acontecer, o que corresponde a dizer que o sistema BASIC ira rejeitar essa linha logo que faca a primeira listagem. Entao algumas instrucoes, senao as restantes linhas do programa, vao desaparecer.

Situacoes como esta devem ser remediadas antes de fazer correr um programa, para evitar perde-lo.

Como por vezes ao elaborarmos uma rotina em codigo maquina nao podemos fugir a situacao de carregarmos um qualquer registo do Z80 com o conteudo 118, uma solucao e a de abandonarmos esta forma de suporte do c.m. (a primeira linha do programa), o que so iremos fazer no proximo capitulo; outra solucao e a de fugirmos ao conteudo 118, arranjando 2 numeros cuja soma seja ele proprio.

Em codigo maquina isto pode ser feito introduzindo no registo o conteudo 117 e incrementando-o (somar 1 ao conteudo).

Em duas rotinas do programa seguinte teremos de recorrer a esse pequeno truque aritmetico.

a) A INSTRUCAO SCROLL

A ROM do ZX vem munida de uma rotina (SCROLL ROUTINE, endereco 3086), que move um bloco do ecran de baixo para cima. No programa anterior "ARTE" convencionamos chamar a essa direccao sentido SUL-NORTE.

A primeira parte da rotina acerta os valores correctos para a posicao PRINT no ficheiro de projeccao, e para a posicao PRINT AT com o numero de linha e coluna.

A segunda parte da rotina prepara o movimento que escreve na linha seguinte o conteudo da anterior.

0C0E SCROLL (ROM)

```
LD B, (DF-SZ)  numero de
          linhas em branco
          fundo do ecran
LD C, 21
CALL 0918, (LOC-ADDR) sub-
               rotina de
        posicao PRINT AT
CALL 099B, (ONE-SPACE)
               subrotina
     que cria um espaco.
LD A, (HL)
LD (DE), A
  LD HL, (D-FILE) endereco
          inicial do fi-
      cheiro de projeccao.
INC HL
LD D, H
LD E, L
CPIR
JP 0A5D, (RECLAIM 1)
```

Para ter bem presente este efeito, vamos escrever um pequeno programa em BASIC.

```
10 REM SCROLL da ROM
 (direccao SUL-NORTE)

20 FOR N=1 TO 200

30 PRINT AT 21,28*RND;"*"

40 SCROLL

50 NEXT N

   RUN
```

Este programa vai deslocar uma serie de asteriscos (*) disseminados ao acaso por uma instrucao RND, ao longo do ecran.

O movimento SCROLL e utilizado em quase todo o tipo de programacao, mas e essencialmente em programas de jogos que o leitor ja tera sentido a necessidade de o executar noutras direccoes do ecran, sem contudo saber como faze-lo.

E possivel, elaborando pequenas rotinas em codigo maquina, obter esse movimento nas restantes direccoes (OESTE-ESTE, ESTE-OESTE e NORTE-SUL) como vamos demonstrar a seguir.

Escreva:

```
  1 REM (30 caracteres)
  2 REM (30 caracteres)
  3 REM (30 caracteres)

 60 REM SCROLL OESTE-ESTE
 65 FOR N = 1 TO 200
 70 PRINT AT 21 * RND, 0; "*"
 80 LET B = USR 16514
 90 NEXT N
 95 STOP
100 REM SCROLL ESTE-OESTE
110 FOR N = 1 TO 200
120 PRINT AT 21 : RND, 31; "*"
130 LET B = USR 16550
140 NEXT N
145 STOP
150 REM SCROLL NORTE-SUL
160 FOR N = 1 TO 200
170 PRINT AT 2, 31 * RND; "*"
180 LET B = USR 16586
190 NEXT N
195 STOP
```

Vamos agora introduzir as 3 rotinas em c.m. nas REMs 1, 2 e 3. Para o fazer sirva-se da ROTINA 3 do programa monitor ou entao escreva uma como esta:

```
500 INPUT X (o primeiro endereco
             util de cada REM)
510 INPUT C (codigo c.m. em
             decimal)
520 IF C > 255 THEN STOP
530 POKE X, C
540 PRINT AT 18, 2 ; X, PEEK X
550 LET X = X + 1
560 GOTO 510
```

(Continua no proximo numero)

```
=========================================================================
```

ENCICLOPEDIA DA LINGUAGEM BASIC

```
=========================================================================
```

AGOSTO/83 - 2

```
---------------
I             I
I   ACS       I
I             I
---------------
```

ESTA FUNCAO E USADA EM basic PARA CALCULAR O arccos DE UM ARGUMENTO n (EM RADIANOS....NAO EM GRAUS)

PODE SER APRESENTADA SOB AS FORMAS : AC. ACSD ACSG ARCOS ARCCOS

MATEMATICAMENTE, DEFINIMOS ARCCOS COMO : angulo A de um triangulo rectangulo formado pela hipotenusa de comprimento H e por um dos lados de comprimento X

A = ARC COS (X/H)

------------------ O OPOSTO A ARCCOS E cosseno de A

o cosseno de um angulo (cuja medida sera A radianos) e o comprimento do lado adjacente ao angulo dividido pela hipotenusa do triangulo rectangulo

COS (A) = X/H

PROGRAMA TESTE

```
10 REM "PROGRAMA TESTE ACS"
20 PRINT "DAR ENTRADA DO VALOR DO cosseno ( de -1 ate 1)";
30 INPUT C
40 LET W = ACS (C)
50 PRINT "O ANGULO COM A RELACAO  X/H DE ";C;" E=";W;" RADIANOS"
60 STOP
```

por exemplo, se usar o valor C = 0.......tera como resultado:

W = 1.5707963 ou arredondando .... W=1.5708 radianos

PARA CONVERTER VALORES DE RADIANOS EM GRAUS, MULTIPLIQUE A MEDIDA DO ANGULO (EM RADIANOS) POR ..... 57.29578, por exemplo :

```
40 LET W = ACS (C) * 57.29578
```

OBSERVACAO

ALGUNS COMPUTADORES CALCULAM O ANGULO EM GRAUS OU GRADOS (100 GRADOS = 90 GRAUS). ESTES COMPUTADORES USAM ..... ACSD PARA GRAUS ..... E ACSG PARA GRADOS

SE O SEU COMPUTADOR NAO POSSUI ESTA FUNCAO, PODERA USAR COMO ALTERNATIVA :

VERIFICAR SE ACEITA .... ATN (arco tangente) e SQR (raiz quadrada)

substituir na linha 40, por :

```
40 LET W = 1.5708 - 2 *ATN (C/(1+SQR (1-C *C)))
```

U N I F I L E (FICHEIRO) ZX81 16K
=============

(Pag. seguinte)

In. YOUR COMPUTER, Abril/83
Trad.e Adapt.de J.MAGALHAES

Este programa para o ZX81 16k permite dar a ficha a estrutura que pretender - OPCAO 1. Por exemplo:

Nome		Morada
Morada	ou	Telefone
Idade		Idade
		Obs.

A OPCAO 2 destina-se ao preenchimento da ficha.
E ainda possivel verificar, alterar ou anular qualquer ficha, devendo para isso dar entrada do nome ou de qualquer outro dado que permita a sua identificacao. Se, por qualquer motivo, nao lhe for possivel seguir esta via, recorra a pesquisa completa do ficheiro desde a primeira ficha, usando "NEWLINE".

* Se por acaso alterou qualquer ficha, nao se esqueca de gravar o programa
* Se modificar a estrutura da ficha, todas as outras serao anuladas
* Se pretender continuar um ficheiro anteriormente gravado, nao utilize a instrucao "RUN" mas "GOTO 40"; Caso contrario perdera todas as outras fichas

OBSERVACAO:
Se com a sua estrutura de ficha o programa para com erro codigo 4/..., tera de redimensionar a matriz B$ na linha 5...Por exemplo:
5 DIM B$(7000)

U N I F I L E

ZX81

16K

```
  1 REM "UNIFILE"
  5 DIM B$(9000)
 40 PRINT AT 0,10;"ficheiro"
 50 PRINT ,,"OPCOES"
 60 PRINT AT 8,0;"    1- ESTRUTURA DA FICHA"
 70 PRINT ,,"    2- ENTRADA DE DADOS"
 80 PRINT ,,"    3- PESQUISA/ALTERAR/ANULAR"
 90 PRINT ,,"    4- ENCERRAR FICHEIRO"
110 SLOW
120 INPUT Z
130 CLS
140 IF Z=1 THEN GOSUB 250
150 IF Z=2 THEN GOSUB 420
160 IF Z=3 THEN GOSUB 1030
170 IF Z=4 THEN GOSUB 200
180 CLS
190 GOTO 40
200 PRINT AT 10,11;"encerrado"
210 PRINT AT 12,6;"* GRAVE O FICHEIRO *"
215 PRINT AT 13,4;"* SE SOFREU ALTERACOES *"
220 PRINT AT 15,9;"(GOTO 2000)"
230 STOP
270 PRINT AT 0,7;"estrutura da ficha"
280 PRINT ,,"NUMERO DE INFORMACOES ?"
290 INPUT X
300 CLS
310 DIM A$(X,20)
320 PRINT AT 0,10;"informacoes"
325 PRINT
330 FOR I=1 TO X
340 PRINT "INFORM. ";I;": ";
350 GOSUB 1570
360 PRINT Q$(2 TO )
370 LET A$(I)=Q$
380 NEXT I
390 LET B$="  "+CHR$ (255)
400 RETURN
440 LET R$=""
450 PRINT AT 0,10;"entradas"
470 PRINT AT 3,0;"""ZZZ"" REGRESSAR AO MENU"
480 PRINT "********************************"
485 PRINT
490 FOR I=1 TO X
500 GOSUB 1600
510 GOSUB 1570
520 PRINT Q$(2 TO )
530 IF Q$="ZZZ" THEN RETURN
540 LET R$=R$+Q$
550 NEXT I
560 CLS
570 LET R$=CHR$ (LEN R$+1)+R$
580 GOSUB 610
590 GOTO 420
630 FAST
640 LET C=2
650 LET C1=C+1
660 DIM C$(100)
670 LET D$=C$
```

```
680 LET D$(1 TO CODE R$(2)-1)=R$(3 TO 1+CODE R$(2))
690 LET C$(1 TO CODE B$(C1)-1)=B$(C1+1 TO C1+CODE B$(C1)-1)
700 IF D$<C$ THEN GOTO 730
710 LET C=C+CODE B$(C)
720 GOTO 650
730 LET B$=B$(1 TO C-1)+R$+B$(C TO )
740 SLOW
750 RETURN
790 LET C2=C2-1
800 LET C1=C+1
810 LET R$=""
820 PRINT "FICHA ";C2;":-"
830 FOR I=1 TO X
840 GOSUB 1600
850 GOSUB 1620
860 PRINT AT 16,0;"********************************"
880 PRINT AT 18,0;">NEWLINE- AVANCA SEM ALTERAR",">""ZZZ""- ANULA A FICHA"
890 GOSUB 1570
900 IF Q$="" THEN LET R$=R$+B$(C1 TO C1+CODE B$(C1)-1)
910 LET C1=C1+CODE B$(C1)
920 CLS
930 IF Q$="" THEN GOTO 960
940 IF Q$="ZZZ" THEN GOTO 970
950 LET R$=R$+Q$
960 NEXT I
970 LET B$=B$(1 TO C-1)+B$(C+CODE B$(C) TO )
980 IF Q$="ZZZ" THEN RETURN
990 LET R$=CHR$ (LEN R$+1)+R$
1000 GOSUB 610
1010 RETURN
1050 LET C=2
1060 LET C4=0
1070 LET C2=1
1080 PRINT AT 0,11;"pesquisa"
1100 PRINT AT 3,0;,,">PESQUISA NORMAL PELO NOME"
1110 PRINT ">NEWLINE- PELA PRIMEIRA FICHA"
1120 PRINT "********************************"
1140 GOSUB 1570
1150 LET S$=Q$
1160 IF S$="" THEN GOTO 1310
1170 IF C+2=LEN B$ THEN RETURN
1180 FAST
1190 LET C1=C+1
1200 IF LEN S$<4 THEN GOTO 1240
1230 IF C4=1 THEN GOTO 1310
1240 FOR I=1 TO X
1250 IF B$(C1 TO C1+CODE B$(C1)-1)=S$ THEN GOTO 1310
1260 LET C1=C1+CODE B$(C1)
1270 NEXT I
1280 LET C=C+CODE B$(C)
1290 LET C2=C2+1
1300 GOTO 1170
1310 LET C1=C+1
1320 IF C+2=LEN B$ THEN RETURN
1330 CLS
1340 SLOW
1350 PRINT "FICHA ";C2;":-"
```

```
1355 PRINT
1360 GOSUB 1640
1370 LET C2=C2+1
1380 PRINT AT 16,0;"********************************"
1400 PRINT ,,">NEWLINE- FICHA SEGUINTE",">""ZZZ""  - REGRESSAR AO MENU",">""AAA"
"  - ALTERAR"
1410 INPUT P$
1420 CLS
1440 IF P$="" THEN GOTO 1310
1450 IF P$<>"AAA" THEN GOTO 1510
1460 LET C=C-C3
1470 LET C4=C
1480 CLS
1490 GOSUB 770
1495 LET C2=1
1500 LET C=2
1510 IF P$="ZZZ" THEN RETURN
1520 CLS
1530 GOTO 1170
1570 INPUT Q$
1580 LET Q$=CHR$ (LEN Q$+1)+Q$
1590 RETURN
1600 PRINT A$(I,2 TO CODE A$(I,1));":";
1610 RETURN
1620 PRINT B$(C1+1 TO C1+CODE B$(C1)-1)
1630 RETURN
1640 FOR I=1 TO X
1650 GOSUB 1600
1660 GOSUB 1620
1670 LET C1=C1+CODE B$(C1)
1680 NEXT I
1690 LET C3=CODE B$(C)
1700 LET C=C+CODE B$(C)
1710 RETURN
2000 SAVE "UNIFILe"
2010 GOTO 10
```

==

JOAO PORTUGAL pretende vender o seu ZX 81

e o modulo 16k por um preco em conta

Contactos: 72225 ou 94105 / COIMBRA

==

ZX81

(Adaptação a outros)

EDUCACAO ...... MATEMATICA.......CALCULO DO POLINOMIO DE LEGENDRE

PROGRAMA EM BASIC ....autor : Dr.Aurelio Fernandes/Aveiro

DESCRITIVO : O Polinomio de Legendre de grau n e dado por :

$$L_n(X) = \frac{2 \times n-1}{n} L_{n-1}(X) \times X - \frac{n-1}{n} \times L_{n-2}(X)$$

em que

$$L_0(X)=1$$

$$L_1(X)=X$$

PROGRAMA

```
  2 REM METODOS NUMERICOS
 10 PRINT "QUAL O GRAU DO POLINOMIO QUE PRETENDE CALCULAR ?"
 20 INPUT I
 30 PRINT "GRAU =";I
 40 DIM L(I+1)
 50 PRINT
 60 PRINT "QUAL O VALOR DA VARIAVEL ?"
 70 INPUT X
 80 PRINT "X = ";X
 90 FOR N=1 TO I+1
100 IF N=1 THEN LET L(N)=1
110 IF N=1 THEN GOTO 150
120 IF N=2 THEN LET L(N)=X
130 IF N=2 THEN GOTO 150
140 LET L(N)=(2*N-1)/N*L(N-1)*X-(N-1)/N*L(N-2)
150 NEXT X
160 PRINT,,,,"PARA X=";X;"E ...N=";I,,,,"O VALOR DO POLINOMIO
E ";L(N-1)
170 PAUSE 500
180 CLS
190 GOTO 10
```

OBSERVACAO : ESTE PROGRAMA FOI ELABORADO NUM ZX81, MAS NAO OFERECE QUALQUER DIFICULDADE DE ADAPTACAO A OUTRA MAQUINA

SPECTRUM
16 ou 48 K

Introduza em primeiro lugar o programa em BASIC da linha 1 a 910, gravando-o do seguinte modo:
SAVE "Reversi" LINE 10.

Agora que tem o programa gravado vai dar inicio a introducao do codigo maquina. Para isso use o comando CLEAR 31999 e de entrada dos 465 bytes com o seguinte programa:

```
10 LET s=0
20 FOR n=32111 TO 32575
30 INPUT b
40 POKE n,b
50 LET s=s+b
60 PRINT n; TAB 10;b; TAB 20;s
70 NEXT n

RUN
```

Depois de ter dado entrada de todo o codigo maquina deve grava-lo desta forma:
SAVE "mcode"CODE 32111,465.
Neste momento, com os dois programas gravados, pode verificar o seu trabalho. Assim inicie o gravador no principio do programa em BASIC, utilizando o comando LOAD "". Nao desligue o gravador ate dar entrada do segundo programa em codigo maquina, sem que para isso seja necessario qualquer comando, ja que o programa em BASIC automaticamente o fara pela instrucao dada na linha 380 LOAD ""CODE.

APRESENTACAO DO JOGO

```
   1 2 3 4 5 6 7 8
 a . . . . . . . . . . a
 b . . . . . . . . . . b
 c . . . . . . . . . . c
 d . . . x o . . . . d
 e . . . o x . . . . e
 f . . . . . . . . . . f
 g . . . . . . . . . . g
 h . . . . . . . . . . h
   1 2 3 4 5 6 7 8
```

A finalidade do jogo e passar todas, ou o maior numero, de pecas do adversario (SPECTRUM) para si, o que lhe e permitido nos seguintes casos:

```
ox        oxo      -     ooo

o          o              o
x          x       -      o
           o              o

o         o              o
 x         x       -      o
            o              o

o         o              o
xx        xx       -     oo
ox        oxo            ooo
```

So pode fazer movimentos em que seja possivel passar pecas do adversario para si. Exemplo:

```
ox o       ou        ox  o
                     xo
```

PROGRAMA "REVERSI" - SPECTRUM 16K (ou 48 K)
LISTAGEM DO PROGRAMA EM BASIC

Reproduzido e corrigido por J.Magalhães
(Your Computer nº 6 Vol 3 Junho83)

```
10 GOTO 380
20 FOR b=1 TO 10: FOR c=1 TO 10: POKE (32000+(b-1)*10+c),a(b,c): NEXT c: NEXT
b: RETURN
30 FOR b=-1 TO 1
40 FOR c=-1 TO 1
50 LET d=m: LET e=n
60 IF a(d+b,e+c)<>s THEN  GO TO 80
70 LET d=d+b: LET e=e+c: GO TO 60
80 IF a(d+b,e+c)<>t THEN  GO TO 120
90 IF a(d,e)=s THEN  LET j=j+1
100 LET a(d,e)=t: IF m=d AND n=e THEN  GO TO 120
110 LET d=d-b: LET e=e-c: GO TO 90
120 NEXT c: NEXT b: RETURN
130 PRINT AT 0,11;"REVERSI"
140 PRINT AT 2,0;"SPECTRUM ""x"" - "
150 PRINT AT 2,16;"TU ""o"" "
160 LET z=4
170 LET g=0: LET h=0
180 PRINT AT (z+1),19;" 12345678 "
190 FOR b=2 TO 9: PRINT AT (b+z),19;CHR$ (b+95);
200 FOR c=2 TO 9
210 IF a(b,c)=x THEN  PRINT "x";
220 IF a(b,c)=o THEN  PRINT "o";
230 IF a(b,c)=f THEN  PRINT ".";
240 IF a(b,c)=x THEN  LET g=g+1
250 IF a(b,c)=o THEN  LET h=h+1
260 NEXT c
280 NEXTbCHR$ (b+95)  270 Print CHR$ (b+95)
290 PRINT AT (z+10),19;" 12345678 "
300 IF g+h=64 THEN  GO TO 810
310 PRINT : PRINT TAB 0;"SPECTRUM ";g;"; TU ";h;".    "
320 RETURN
330 FOR b=1 TO 10: FOR c=1 TO 10
340 IF b<>1 AND c<>1 AND b<>10 AND c<>10 THEN  LET a(b,c)=f
350 NEXT c: NEXT b
360 LET a(5,5)=x: LET a(6,6)=x: LET a(6,5)=o: LET a(5,6)=o
370 RETURN
380 CLEAR 31999: POKE 23609,75: LOAD ""CODE : CLS
390 PRINT AT 11,0; FLASH 1;"Desligue o gravador"
400 DIM a(10,10): LET z$="                               ": LET f=46: LET j=0: LET k=48: LET o
=111: LET r=0: LET u=57: LET v=k+k: LET w=k+u: LET x=120
410 PRINT AT 20,0;"Qualquer tecla para comecar": IF INKEY$="" THEN  GO TO 410
420 CLS : PRINT AT 11,0;"Aguarde um momento": GO SUB 330: GO SUB 20
430 CLS : PRINT AT 10,3;"Este jogo tem dois niveis.";AT 13,3;"(ENTER 1) - Facil
             (ENTER 2) - Dificil"
440 INPUT l: IF l<1 OR l>2 THEN  GO TO 440
450 IF l=1 THEN  POKE 32528,0: POKE 32529,0: POKE 32530,0: GO TO 470
460 POKE 32528,205: POKE 32529,173: POKE 32530,125
470 CLS : PRINT AT 0,12;"REVERSI"
480 PRINT AT 2,2;"Para o teu movimento, da          entrada da letra seguida
  do num.  Ex: ""d4"" - ""b8"""
```

```
490 LET z=6: GO SUB 170
500 INPUT "Queres jogar primeiro? (S/N)";q$
510 CLS : GO SUB 130
520 IF q$="s" OR q$="S" THEN  GO TO 640
530 PRINT AT 11,2;"  SPECTRUM   ";AT 12,0;z$;AT 13,0;z$
540 LET s=o: LET t=x
550 RANDOMIZE
560 LET i=USR 32445
570 IF i=0 AND r=0 THEN  GO TO 810
580 IF i=0 THEN  PAUSE 50: BEEP .25,25: BEEP .25,20: PRINT AT 10,5;"Nao posso";
AT 11,5;"jogar": PAUSE 75: GO TO 640
590 LET i=i-32000
600 LET m=INT (i/10)
610 LET n=i-m*10: LET m=m+1
620 BEEP .25,25: BEEP .25,20: PRINT AT (4+m),(n+18); FLASH 1;"x"
630 GO SUB 30: GO SUB 20: BEEP .25,25: BEEP .25,15: GO SUB 160
640 PRINT AT 11,2;"TEU MOVIMENTO";AT 7,0;z$;AT 6,0;"""Z9""-Se impedido";AT 7,6;
"de jogar"
650 LET s=x: LET t=o
660 INPUT p$: IF LEN p$<>2 THEN  PRINT AT 19,0; FLASH 1;"Entrada de 2 caractere
s nao ";LEN p$;"    ": GO TO 660
670 IF p$="z9" OR p$="9z" THEN  LET r=0: GO TO 750
680 IF (CODE p$>v AND CODE p$<w AND p$(2)>CHR$ k AND p$(2)<CHR$ u) OR (CODE p$>
k AND CODE p$<u AND p$(2)>CHR$ v AND p$(2)<CHR$ w) THEN  GO TO 700
690 PRINT AT 19,0; FLASH 1;"Nao podes jogar ";p$;" Tenta de novo": PRINT z$;z$:
 GO TO 660
700 IF CODE p$>k AND CODE p$<u AND p$(2)>CHR$ v AND p$(2)<CHR$ w THEN  LET p$=p
$(2)+p$(1)
710 LET m=CODE p$-95: LET n=VAL p$(2)+1: LET r=(m-1)*10+n-1
720 IF a(m,n)<>f THEN  GO TO 800
730 LET j=0
740 GO SUB 30: IF j<1 THEN  GO TO 790
750 PRINT AT 19,0; FLASH 0;z$;z$
760 PRINT  FLASH 0;z$;z$: IF r=0 THEN  GO TO 530
770 BEEP .25,25: BEEP .25,20: PRINT AT (4+m),(n+18); FLASH 1;"o"
780 GO SUB 20: BEEP .25,25: BEEP .25,15: GO SUB 160: GO TO 530
790 LET a(n,m)=f
800 PRINT AT 19,0; FLASH 1;"Nao podes jogar ";p$;" tenta de novo"; FLASH 0;" ":
PRINT  FLASH 0;z$;z$: GO TO 660
810 PRINT AT 10,0;z$;AT 12,0;z$;AT 13,0;z$;AT 16,0;z$;z$
820 IF g>h THEN  PRINT AT 18,0;"Ganhei, ";g;" - ";h
830 IF g<h THEN  PRINT AT 18,0;"Perdi, ";h;" - ";g
840 IF g=h THEN  PRINT AT 18,0;"E um desenho: ";g;" - ";h
850 INPUT "Pretendes novo jogo? (S/N)";y$
860 IF y$="n" OR y$="N" THEN  STOP
870 CLS : PRINT AT 11,0;"Aguarde um momento": GO SUB 330: GO SUB 20
880 CLS : INPUT "Escolha o nivel (1 ou 2)? ";l
890 IF l<1 OR l>2 THEN  GO TO 880
900 IF l=1 THEN  POKE 32528,0: POKE 32529,0: POKE 32530,0: GO TO 500
910 POKE 32528,205: POKE 32529,173: POKE 32530,125: GO TO 500
```

listagem do codigo maquina

```
PROGRAMA REVERSI......SPECTRUM.....(CONTINUACAO)
32111 62 50 190 208 62 115 190 56 8 58
32121 107 125 60 50 107 125 201 58 106 125
32131 33 107 125 134 50 106 125 201 62 0
32141 50 107 125 125 145 111 205 111 125 62
32151 111 190 40 245 201 62 0 50 107 125
32161 125 129 111 205 111 125 62 111 190 40
32171 245 201 125 254 89 202 134 126 254 82
32181 202 134 126 254 19 202 134 126 254 12
32191 202 134 126 254 87 202 142 126 254 86
32201 202 142 126 254 85 202 142 126 254 84
32211 202 142 126 254 69 202 142 126 254 62
32221 202 142 126 254 59 202 142 126 254 52
32231 202 142 126 254 49 202 142 126 254 42
32241 202 142 126 254 39 202 142 126 254 32
32251 202 142 126 254 17 202 142 126 254 16
32261 202 142 126 254 15 202 142 126 254 14
32271 202 142 126 254 88 40 126 254 83 40
32281 122 254 79 40 118 254 78 40 114 254
32291 77 40 110 254 76 40 106 254 75 40
32301 102 254 74 40 98 254 73 40 94 254
32311 72 40 90 254 68 40 86 254 63 40
32321 82 254 58 40 78 254 53 40 74 254
32331 48 40 70 254 43 40 66 254 38 40
32341 62 254 33 40 58 254 29 40 54 254
32351 28 40 50 254 27 40 46 254 26 40
32361 42 254 25 40 38 254 24 40 34 254
32371 23 40 30 254 22 40 26 254 18 40
32381 22 254 13 40 18 58 106 125 201 58
32391 106 125 203 39 203 39 201 58 106 125
32401 203 39 201 58 106 125 60 203 47 201
32411 237 75 118 92 205 43 45 239 161 15
32421 52 55 22 4 52 128 65 0 0 128
32431 50 2 161 3 49 56 205 162 45 237
32441 67 118 92 201 62 0 50 105 125 50
32451 108 125 50 109 125 6 78 33 11 125
32461 120 133 111 62 46 190 32 100 229 62
32471 0 50 106 125 14 11 205 139 125 225
32481 229 205 156 125 225 229 13 205 139 125
32491 225 229 205 156 125 225 229 13 205 139
32501 125 225 229 205 156 125 225 229 14 1
32511 205 139 125 225 229 205 156 125 225 229
32521 58 106 125 254 0 40 40 0 0 0
32531 33 105 125 190 56 31 32 17 245 197
32541 229 205 155 126 33 0 78 167 237 66
32551 56 12 225 193 241 50 105 125 225 34
32561 108 125 24 4 225 193 241 225 16 143
32571 237 75 108 125 201 0 0 0 0 0
```

INTRUCOES PARA O PROGRAMA

OTHELLO (Fernando Preces)

publicado no numero 8 - Maio

Comandos para jogar

As teclas de movimento, marcadas com as respectivas setas direccionais (teclas 5, 6, 7 e 8) deslocam os dois apontadores ↓ → que aparecem no jogo a direita do tabuleiro, todas as vezes que e o utilizador a jogar.
A intercepcao desses dois apontadores marca as coordenadas, ponto aonde o SPECTRUM ira colocar a nossa pedra quando for premida a tecla "0".

Regra fundamental do jogo

Uma pedra somente pode ser colocada num ponto estrategico, tal que combata e aniquile a posicao do adversario.
Para o SPECTRUM aceitar a posicao e necessario que em oposicao exista ja uma pedra da mesma cor. O SPECTRUM esta programado para as regras do jogo e NAO ACEITA marcacao de posicoes que fujam a esta atribuicao.

F I C H E I R O D E E N D E R E C O S

(pag. seguinte)

JORGE AGUIAR/Porto

Este programa apresenta um ficheiro de nomes, enderecos e numeros de telefone.
Contem um menu com 6 opcoes, 5 das quais sao usadas para processamento de dados.
Cada ficha contem 48 espacos; desses 48, 20 sao usados para o nome, 20 para a morada e 8 para o numero de telefone.
No total, podem ser abertas 100 fichas.

O ficheiro pode ser consultado a qualquer momento atraves da segunda opcao do menu. Esta funciona dando entrada a um nome, procurando-o no ficheiro e desenhando um quadro com o nome, a morada, etc. Esta opcao tambem permite obter uma copia da ficha que esta no ecran.

E tambem possivel alterar qualquer ficha ja registada. A terceira opcao permite-o, dando entrada a um nome. Entao o computador ira procurar a ficha com o nome indicado e tudo prosseguira como se V. estivesse a escrever novamente essa ficha.

A quarta opcao mostra todo o ficheiro, dando a possibilidade de copiar as fichas para a ZX PRINTER.

Finalmente, a quinta opcao pode apagar qualquer ficha, dando entrada a um nome, procurando-o no ficheiro e puxando para a posicao anterior todas as fichas que estiverem nas posicoes seguintes a da ficha pedida.

Pode gravar o ficheiro actual, utilizando a sexta opcao, tal como pode chamar algum ficheiro gravado anteriormente por este programa, e com um nome escolhido por si.

programa FICHEIRO DE ENDERECOS
SPECTRUM 48 K
autor....JORGE AGUIAR/PORTO

```
  50>REM "TELEFONE":REM Producao exclusiva de Jose Jorge
  60 LET z=0: DIM a$(100,48)
 100 BORDER 0: PAPER 0: INK 7: CLS : LET z$=""
 110 PRINT AT 1,13; FLASH 1;"OPCOES": PRINT AT 2,13;"______"
 120 PRINT AT 3,1;"1-Abrir uma ficha"
 130 PRINT AT 5,1;"2-Consultar o ficheiro"
 140 PRINT AT 7,1;"3-Alterar uma ficha"
 150 PRINT AT 9,1;"4-Verificar todo o ficheiro"
 160 PRINT AT 11,1;"5-Apagar uma ficha"
 161 PRINT AT 13,1;"6-Gravar ou chamar dados"
 170 PRINT AT 17,1; INK 0; PAPER 7; FLASH 1;"ATENCAO";: PRINT ":o ficheiro so co
mporta          100 fichas."
 180 PRINT AT 21,1;"Escreva o numero da sua opcao.";: PRINT  FLASH 1;"*"
 190 LET z$=INKEY$: IF z$<"1" OR z$>"7" THEN  GO TO 190
 200 GO TO VAL z$*1000
1000 BORDER 7: PAPER 7: INK 0: CLS
1010 LET z=z+1: PRINT TAB 4; INK 0;"ABRIR UMA FICHA": PRINT
1020 GO SUB 6500
1025 LET a$(z)( TO 20)=b$
1027 PRINT : PRINT "      Agora a morada,"
1030 GO SUB 6520
1035 LET a$(z)(21 TO 40)=b$
1040 PRINT : PRINT "      Agora o telefone."
1050 INPUT  LINE b$: IF LEN b$>8 OR b$="" THEN  GO TO 1050
1060 LET a$(z)(41 TO 48)=b$
1070 GO TO 100
2000 BORDER 7: PAPER 7: INK 0: CLS
2010 PRINT TAB 3; INK 0;"CONSULTAR O FICHEIRO": PRINT
2020 GO SUB 6500: GO SUB 7000: IF i=1 THEN  GO TO 9000
2030 GO SUB 8000: GO TO 100
3000 REM ALTERACOES
3010 BORDER 7: PAPER 7: INK 0: CLS
3020 PRINT TAB 3; INK 0;"ALTERAR UMA FICHA": PRINT
3030 GO SUB 6500: GO SUB 7000: IF i=1 THEN  GO TO 9000
3040 GO SUB 8000: CLS : PRINT "Agora pode altera-la.": PRINT
3050 GO SUB 6500: LET a$(n)( TO 20)=b$: PRINT : PRINT "      Agora a morada,"
3060 GO SUB 6520: LET a$(n)(21 TO 40)=b$: PRINT : PRINT "      Agora o telefone."
3070 INPUT  LINE b$: IF LEN b$>8 OR b$="" THEN  GO TO 3070
3080 LET a$(n)(41 TO 48)=b$: PRINT
3090 PRINT "Quer ver como ficou?";: PRINT  FLASH 1;"(S/N)"
3100 LET z$=INKEY$: IF z$<>"s" AND z$<>"n" THEN  GO TO 3100
3110 IF z$="n" THEN  GO TO 100
3120 GO SUB 8000: GO TO 7
4000 BORDER 7: PAPER 7: INK 0: CLS
4010 FOR n=1 TO 100
4020 IF a$(n)( TO 20)="                    " THEN  GO TO 4050
4030 GO SUB 8000
4040 NEXT n
4050 CLS : PRINT "E tudo.Quer ver outra vez?";: PRINT  FLASH 1;"(S/N)"
4060 LET z$=INKEY$: IF z$<>"s" AND z$<>"n" THEN  GO TO 4060
4070 IF z$="s" THEN  GO TO 4000
4080 GO TO 100
5000 BORDER 7: PAPER 7: INK 0: CLS
5010 PRINT TAB 3; INK 0;"APAGAR UMA FICHA": PRINT
5015 GO SUB 6500
5020 GO SUB 7000
5030 IF i=1 THEN  GO SUB 9000
5040 GO SUB 8000
5045 CLS
5050 FOR v=n+1 TO z+1
5060 LET a$(v-1)( TO 48)=a$(v)( TO 48)
5070 NEXT v
```

Programa " Ficheiro de endereços " SPECTRUM (continuação)

```
5090 LET z=z-1
5100APAGADA  5100 Print "Ficha eliminada "
5110 INPUT "Prima ~ENTER~."; LINE z$
5120 GO TO 100
6000 BORDER 7: PAPER 7: INK 0: CLS
6020 PRINT TAB 3; INK 0;"GRAVAR OU CHAMAR DADOS": PRINT
6030 PRINT "Quer gravar ou chamar dados?(G/C)"
6040 LET z$=INKEY$: IF z$<>"g" AND z$<>"c" THEN  GO TO 6040
6050 IF z$="c" THEN  GO TO 6210
6060 CLS : PRINT TAB 1;"Meta na entrada ~MIC~ do grava- dor o cabo ligado a entr
ada        ~MIC~ do computador": PRINT : PRINT
6070 PRINT "Pronto?": INPUT "Prima ~ENTER~ quando pronto."; LINE z$
6080 PRINT AT 0,0;"                                 "
6085 INPUT "Qual e o nome que quer dar aos dados?"; LINE z$
6090 SAVE "TELEFONE" DATA a$()
6100 CLS : PRINT  FLASH 1;"PARE O GRAVADOR!"
6110 IF INKEY$="" THEN  GO TO 6110
6120 CLS : PRINT "Recue a ~Cassete~ ate onde come-cou a gravacao."
6130 IF INKEY$="" THEN  GO TO 6130
6140 PRINT : PRINT "Agora meta na entrada ~EAR~ do  computador o cabo ligado na
en- trada ~EAR~ do gravador.": PRINT
6150 IF INKEY$="" THEN  GO TO 6150
6160 PRINT "Ponha o gravador a tocar normal-mente e prima ~ENTER~."
6170 INPUT  LINE z$
6175 CLS
6177 PRINT "Caso o programa parar e escre-  ver ~R Tape Loading Error~ p/f/  esc
rever ~GO TO 100~ e gravar o  programa novamente."
6180 VERIFY z$ DATA a$()
6190 CLS : PRINT "Pronto.Prima ~ENTER~."
6200 GO TO 100
6210 CLS : PRINT TAB 1;"Meta na entrada ~EAR~ do grava- dor o cabo ligado a entr
ada        ~EAR~ do computador": PRINT : PRINT
6220 IF INKEY$="" THEN  GO TO 6220
6230 INPUT "Qual e o nome dos dados desejados?"; LINE z$
6240 CLS
6250 LOAD z$ DATA a$()
6260 GO TO 100
6500 PRINT
6510 PRINT "      Escreva o nome,"
6520 PRINT "sem exceder 20 caracteres."
6530 INPUT  LINE b$: IF LEN b$>20 THEN  GO TO 6530
6540 RETURN
7000 PRINT
7010 FOR n=1 TO 100: LET i=0
7020 IF a$(n)( TO LEN b$)=b$ THEN  GO TO 7050
7030 NEXT n
7040 LET i=1
7050 RETURN
8000 CLS
8010 PLOT 0,170: DRAW 250,0: DRAW 0,-170: DRAW -250,0: DRAW 0,170
8020 PLOT 13,0: DRAW 0,170
8030 CIRCLE 6,17,3: CIRCLE 6,52,3: CIRCLE 6,87,3: CIRCLE 6,122,3: CIRCLE 6,157,3
8040 PLOT 13,133: DRAW 238,0: PLOT 13,109: DRAW 238,0
8050 PLOT 13,69: DRAW 238,0: PLOT 13,33: DRAW 238,0
8055 OVER 1
8060 PRINT AT 1,10;"FICHA No.";n
8070 PRINT AT 6,2;"NOME:";a$(n)( TO 20)
8080 PRINT AT 10,2;"MORADA:";a$(n)(21 TO 40)
8090 PRINT AT 15,2;"No.TELEFONE:";a$(n)(41 TO 48)
8100 PRINT AT 19,2; INK 7; PAPER 0;"FICHEIRO DE TELEFONE"
8110 PRINT AT 20,3;"  1983 by Jose Jorge"
```

PROGRAMA "FICHEIRO DE ENDEREÇOS/TELEFONES" ...... continuação

FICHA No.1
NOME:CLUBE Z 80
MORADA:AV.BOAVISTA 832/2-T
No.TELEFONE:65127
FICHEIRO DE TELEFONE © 1983 by Jose Jorge

```
8115>OVER 0
8120 INPUT "Quer copiar?(S/N)"; LINE z$
8121 IF z$<>"s" AND z$<>"n" THEN  GO TO 8120
8122 IF z$="s" THEN  COPY
8130 RETURN
9000 REM NAO EXISTE
9010 PRINT  FLASH 1;"ESSE NOME NAO EXISTE NO FICHEIRO": FOR n=10 TO 8 STEP -1
9020 BEEP .2,n: NEXT n
9030 BEEP .4,7: PAUSE 70: GO TO 100
```

observacoes
----

acrescentar as seguintes linhas

```
1001>IF z<100THEN GO TO 1010
1002 FLASH 1: PRINT "ja existem  100 FICHAS"
1003 PRINT AT 18,0;"NAO EXISTE ESPACO NO FICHEIRO"
1004 FOR N=10 TO 0 STEP -1: BEEP .2,N: NEXT N: BEEP .4,7
1005 INPUT "usar ENTER"; LINE Z$
```

NO PROXIMO NUMERO INICIAREMOS UMA NOVA RUBRICA DA AUTORIA DE FERNANDO PRECES:

```
******************
*   E S P A C O   *
* S P E C T R U M *
******************
```

R E L O G I O (COM PONTEIROS)

NEWBRAIN

ISABEL CRISTINA / Porto

```
10  REM RELOGIO
20  CLOSE#1: OPEN#0,0,"130"
30  OPEN#1,11,"w160"
40  Plot range(200,100),centre(50,50)
50  Plot degrees
60  Plot background(1),wipe
70  PUT 23,65
90  REM
100 Plot Place(50,50),move(50,-50)
110 Plot Place(-20,-45),"Segundos"
120 Plot Place(82,-45),"Minutos"
140 REM
150 Plot Place(-10,0),turn(90)
160 FOR D = 5 TO 60 STEP 5
170   Plot turn(90-(D*6)),colour(2)
180   Plot moveby(35),colour(1),D
190   Plot Place(-10,0)
200 NEXT D
220 REM
230 Plot centre(150,50)
240 Plot Place(-10,0),turn(90)
250 FOR D = 5 TO 60 STEP 5
260   Plot turn(90-(D*6)),colour(2)
270   Plot moveby(35),colour(1),D
280   Plot Place(-10,0)
290 NEXT D
320 REM
330 M=96: Plot centre(150,54)
340 Plot Place(0,0),turn(M)
350 Plot drawby(25,2): M = M - 6
360 Plot turn(M),drawby(25,1)
380 REM
390 Plot centre(50,54),Place(0,0)
400 Plot turn(90)
410 FOR S = 1 TO 60
420   Plot drawby(25,2),turnby(-6)
430   Plot drawby(25,1)
440   GOSUB 500
450 NEXT S
460 Plot centre(150,54)
470 GOTO 340
500 REM
510 FOR X = 1 TO 270
520 NEXT X
530 RETURN
```

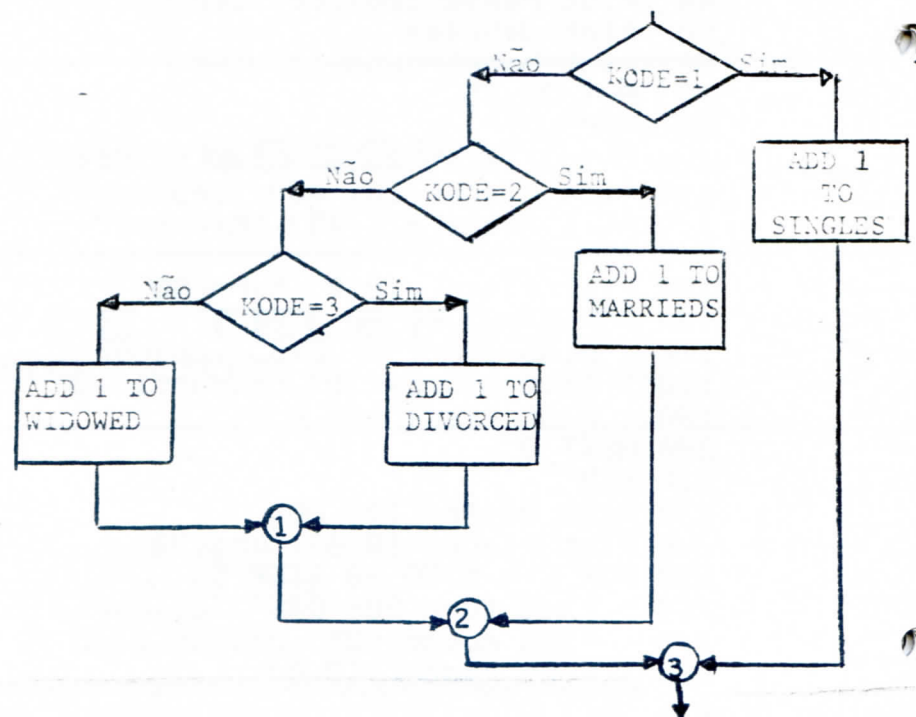
==

P R O G R A M A C A O   E S T R U T U R A D A
(continuacao)

==

in. "STUCTURED COBOL", PHILIPPAKIS Andreas e KAZMIER Leonard

Traduzido e adaptado ao BASIC por
FRANCISCO J.R. SANTOS

A convergencia de todas as frases de GO TO para NEXT-P e compativel com os principios da programacao estruturada, em que cada trabalho estruturado deve ter um ponto de inicio e um ponto de fim. Repare-se tambem que GO·TO e necessario nesta estrutura. Contudo, de modo a preservar a propria estrutura, o paragrafo referenciado pelo GO TO deve ser localizado fisicamente a seguir aonde acabam as frases de GO TO.
A estrutura CASE pode ser tambem executada em termos de encaixar estruturas IF-THEN-ELSE como exemplificado no ordinograma seguinte.
Repare-se que cada IF-THEN-ELSE comeca e acaba marcado dentro de um circulo, para haver uma maior clareza. Assim, por exemplo, cada estrutura IF-THEN-ELSE termina por um circulo numerado de 1 a 4; Para melhor elucidar o exemplo:

ORDINOGRAMAS ESTRUTURADOS

Tradicionalmente, os ordinogramas tem sido usados por dois objectivos: o primeiro para ajudar o programador a desenvolver um programa logico; o segundo para servir de documentacao a um programa completo. Nos ultimos anos tem havido uma tendencia para minimizar e mesmo por de parte o uso dos ordinogramas. O caso e que, como tem sido feito, se o programador utiliza os principios de uma boa programacao estruturada, o programa ja por si so e um documento e portanto o ordinograma e superfluo.
Ha aqueles que afirmam que num meio onde

as alteracoes de programas sao frequentes, tendo para isso que se desenhar sucessivamente novos ordinogramas, isso se torna macador e dispendioso. Discordamos com estes que, de um modo radical, afirmam que os ordinogramas sao inuteis. Em vez disso, vemos que os ordinogramas sao uma ferramenta util para o programador e sua posterior documentacao, e reconhecemos que o uso deste auxiliar da programacao nao devera ser indiscriminado. Em qualquer caso, para o principiante, o uso de ordinogramas e de grande valor. (...) Um ordinograma estruturado tem que ter estas caracteristicas principais:

- ser claro
- ser simples e compreensivel
- ter um principio e um fim claros
- utilizar as cinco estruturas principais
  - *sequencia
  - *if-then-else
  - *do-while
  - *do-until
  - *case

LOGICA DA PROGRAMACAO ESTRUTURADA

Ja temos visto que a programacao estruturada depende bastante da habilidade de dividir um trabalho complexo em trabalhos simples e pequenos, cujos detalhes podem ser abstraidos.

Cada parte de um programa estruturado e tratado como um modulo. O modulo por si so e uma estrutura de programa. Tem uma entrada e uma saida e le-se naturalmente de cima para baixo, podendo consistir em uma das cinco estruturas basicas do programa (sequencia, if-then-else, do until, do while e case). O modulo e referenciado por um nome, o qual deve ser o mais descritivo possivel. Em BASIC, esse nome e um nome de paragrafo ou um nome de seccao. Um modulo nao deve exceder uma pagina de comprimento, para poder ser facilmente lido. Finalmente, cada modulo e executado por uma referencia de uma parte para outra do programa, geralmente atraves do verbo GO SUB.

A figura 1 apresenta em linhas gerais um programa estruturado.

Figura 1

```
Begin Main Program Module

    Instruction 1

    ...

    ...

    ...

    ...

    ...

    Instruction M

    Execute Module A

    Execute Module B

    ...

    ...

    Instruction n

End Main Program Module


Begin Module A

    Instruction 1

    Execute Module X

    ...

    ...

    Instruction n
```

```
End Module A


Begin Module B

    Instruction 1

    ...

    ...

    ...

    Instruction n

End Module B


Begin Module X

    Instruction 1

    ...

    ...

    ...

    Instruction n

End Module X
```

De notar que ha um modulo principal e um numero de modulos dependentes. A parte do modulo principal da um sumario do programa - e e geralmente bastante pequeno - em que o essencial do programa e facilmente percebido no contexto do modulo principal (Main Program Module). Repare que o modulo A e executado sob o controlo do modulo principal. Se o nome do modulo principal for significativo, nao precisamos de nos incomodar com os detalhes do modulo A. Pode muito bem ser o caso que, como no exemplo, o modulo A e suficientemente grande ou complexo que precisamos dividi-lo no modulo X, fora do modulo A. Assim, o modulo e executado sob o controle do modulo A, que por sua vez e executado sob o comando do modulo principal. Assim a interligacao destes modulos e permissivel e desejavel.

Precisamos so de ter em mente que assim que se alcance o fim do modulo, se retome o modulo que assegura a execucao das instrucoes. Assim, quando o modulo e executado voltamos a instrucao 2 do modulo A; e assim que o modulo A e executado segue-se para EXECUTE MODULE B, instrucao do modulo principal.
Torna-se agora obvio que o verbo Perform e exactamente o mecanismo de linguagem que se presta a este tipo de estrutura e ao controlo de execucao.
De salientar que esta estrutura basica tem sido usada em programas exemplo, ja discutidos aqui. EXHIBIT-1 apresenta um excerto de um programa estruturado e da uma oportunidade de observar a implementacao de conceitos da estrutura do programa, discutidos neste capitulo. A natureza especifica do trabalho, como e claro, nao e discutida neste ponto.
Ha, contudo, duas condicoes que restringem o seu uso. Alguns trabalhos sao comuns a mais que um programa. Repetir a mesma instrucao em cada um desses programas e indesejavel.
E dificil para os membros de uma equipa de programadores trabalharem simultaneamente no mesmo programa. E facil subdividir o trabalho composto em programas independentes, constituindo cada um um modulo, que possam ser escritos, compilados e testados por programadores trabalhando individualmente e independentemente dos outros. Este ultimo objectivo e conseguido usando subrotinas ou subprogramas.
Um programa subrotina e um programa tal como qualquer outro programa, excepto quando nao e executavel por ele proprio, ou seja, que so pode ser executado sob o controlo de outro programa. Deve ser claro, contudo, que com ou sem uso de programas subrotinas, o programador de BASIC pode criar programas bem estruturados e modulares.

FIM

========================

SOFTWARE S P E C T R U M

========================

PROGRAMA : SHIP OF DOOM
MAQUINA : SPECTRUM 48 K

TIPO DE PROGRAMA : JOGO DE AVENTURAS C/EMOCAO E ACCAO

DESCRITIVO : O utilizador aterrou num planeta com uma avaria na nave.Verifica que o planeta e habitado e que existe um edificio sombrio, onde tera de se aventurar.
Vai a procura de auxilio para reparar a nave, mas o perigo espreita na sombra;por isso tera de lutar para sobreviver

PODE DIALOGAR COM A MAQUINA, USANDO PALAVRAS EM INGLES TAIS COMO : take; break; cut; with,...etc

Depois de entrar no edificio, devera orientar-se de acordo com os pontos cardeais N - S - E - W

O JOGO TERMINA QUANDO SE DESCOBREM TODOS OS SEGREDOS DO EDIFICIO E SE ENCONTRAM AS PECAS PARA REPARAR A NAVE, DEVENDO ENTAO REGRESSAR PELO MESMO CAMINHO.

TRATA-SE DE UM JOGO DE PALAVRAS

Preço/Clube Z80...35o$

****
****

PROGRAMA : (CORRIDA DE CARACOIS)
MAQUINA : SPECTRUM (16K ou 48 K)

TIPO DE PROGRAMA : JOGO DE CORRIDAS COM APOSTAS, ATE 6 UTILIZADORES

DESCRITIVO : Neste jogo os caracois partem todos em linha, e um deles vai ficar doente.

OS APOSTADORES PODEM APOSTAR NO MESMO CARACOL, ATE UM MAXIMO DE 100$00 OU EM DIFERENTES CARACOIS

Preço/Clube Z80...35o$

****
****

========================

SOFTWARE S P E C T R U M

========================

PROGRAMA : POKER

MAQUINA : SPECTRUM 16 K ou 48 K

TIPO DE PROGRAMA : JOGO DE CARTAS C/TODAS AS NUANCES DO POKER DE CARTAS

DESCRITIVO : PODE JOGAR COM O COMPUTADOR, SENDO ELE O SEU ADVERSARIO

O VALOR DE QUE DISPOE PARA APOSTAR E DE $10 NO MAXIMO E SERA SEMPRE OBRIGADO A IR AO JOGO (minimo de aposta $1)

Tera no ecran um grupo de 5 cartas, e podera pedir para trocar qualquer uma delas

As combinacoes possiveis sao :

```
SEQUENCIA MAX. (A,K,Q,J,10)
POKER      (K,K,K,K)
FULL HOME (D,D,9,9,9)
FLUSH      (A,J,9,K,9)
SEQUENCIA MIN.(K,Q,J,10,9)
TRIO       (J,J,J)
2 PARES    (A,A,D,D)
1 PAR      (9,9)
CARTA + ALTA
```

OBSERVACAO

AS CARTAS APARECEM NO ECRAN, DESENHADAS COM OS SEUS NAIPES RESPECTIVOS

Preço/Clube Z8o...350$

****
****

PROGRAMA : MORSE

MAQUINA : SPECTRUM (16 K ou 48 K)

TIPO DE PROGRAMA : DESTINA-SE A APRENDIZAGEM E PRATICA DE CODIGO MORSE

DESCRITIVO :Podera regular a velocidade de transmissao. Tera as opcoes de praticar com caracter a caracter ou com frases.

Permitir-lhe-a escrever em portugues e escutar os sons respectivos (do codigo) ou inversamente tentar traduzir para codigo MORSE os caracteres ou frases que o computador selecciona de modo aleatorio.

Preço/Clube Z8o...400$00

C L U B E Z - 80

****

I N S C R I C A O COMO ASSOCIADO

****

O CLUBE Z-80 esta aberto a todos os utilizadores de microcomputadores.

A intencao de associar os entusiastas das micro-maquinas, e exclusivamente a de permitir :

1 - PUBLICACAO DE UM JORNAL MENSAL, onde sejam publicados programas de uso geral ou especifico como no caso da educacao.

2 - PROMOVER TROCAS DE PROGRAMAS, e trocas de experiencias ; tanto no caso do Software(programacao), como no caso do Hardware (electronica).

3 - PROMOVER DESCONTOS NA AQUISICAO DE PROGRAMAS

4 - LANCAR CURSOS DE PROGRAMACAO EM BASIC - PASCAL OU OUTRAS LINGUAGENS E DIVULGAR O USO DE LINGUAGEM MAQUINA

==

CLUBE Z-80
AV DA BOAVISTA, 832-2 ° T
TELEFONE 65127
4100 PORTO • **PORTUGAL**

NOME ....

IDADE .... COMPUTADOR TIPO ....

PROFISSAO...........................

ENDERECO...
....
....

ASSINATURA ANUAL -------esc. 1500$00 ☐

ASSINATURA SEMESTRAL ---- esc. 750$00 ☐

CHEQUE OU VALE DO CORREIO
Nº.............................BANCO.............................

DATA ...../..../.....

JÁ SÓCIO ☐

NOVO SÓCIO ☐ → A partir do mês de ____________(inclusivé)

CARLOS OLIVEIRA
ASSINATURA ANUAL DESDE SETEMBRO/82.
CADUCA NO FIM DESTE MES. SE PRETENDE RENOVAR, UTILIZE ESTE CUPAO.